

File created: 3-Apr-86 21:33:56 {LOGOS:AFB:SIP}<DOUG>LISP>ACCESS.;3

changes to: (VARS ACCESSCOMS)
(FNS ACCESS)

previous date: 1-Apr-86 17:15:26 {LOGOS:AFB:SIP}<DOUG>LISP>ACCESS.;2

Read Table: OLD-INTERLISP-FILE

Package: INTERLISP

Format: XCCS

(* * Copyright (c) 1986 by Speech Input Project, Univ. of Edinburgh.
All rights reserved.)

```
(RPAQQ ACCESSCOMS ((FNS ACCESS ACCESS.PARSE ACCESS.PARSE.DIRECTORY ACCESS.DIRECTORYNAME ACCESS.MKDIR
ACCESS.PARSE.OBJ ACCESS.PARSE.ACCESS ACCESS.DO.COMMAND ACCESS.OPEN ACCESS.SHOW
ACCESS.ADD ACCESS.REMOVE ACCESS.CHANGE ACCESS.SLASHIFY.DIRNAME ACCESS.NUM.TO.STRING
ACCESS.STRING.TO.NUM)))
```

(DEFINEQ

ACCESS

[LAMBDA NIL

(* drc: " 3-Apr-86 21:32")

(* * Top level function. Parses a command line, checks args and performs commands.)

```
(bind (USER/PWD _ (\INTERNAL/GETPASSWORD NIL NIL NIL NIL NIL 'NS))
COM DIR OBJ ACCESS VALUE do [SETQ VALUE (NLSETQ (LET ((COMMAND (ACCESS.PARSE DIR OBJ ACCESS USER/PWD)
))
(SELECTQ (SETQ COM (CAR COMMAND))
(ABORT (PRINTOUT T "[aborted]"))
(Q (TERPRI T))
(L (SETQ USER/PWD (CDR COMMAND)))
(P (TERPRI T)
(USEREEXEC '___))
(PROGN (for X in (CDR COMMAND) as Y
in ' (DIR OBJ ACCESS)
do (SET Y X))
(ACCESS.DO.COMMAND USER/PWD COM DIR
OBJ ACCESS]
[if (NLISTP VALUE)
then (LET ((ERROR (ERRORN)))
(* printout error messages, but treat CONROL-E errors as
aborts)
(if (EQ (CAR ERROR)
47)
then (PRINTOUT T "[aborted]")
else (ERRORMESS ERROR)
until (EQ COM 'Q])
```

ACCESS.PARSE

[LAMBDA (DIR.DEFAULT OBJ.DEFAULT ACCESS.DEFAULT USER/PWD)

(* drc: " 1-Apr-86 17:09")

(* * Prompts for and parses one command line. Simulates ASKUSER.)

```
(RESETFORM (CONTROL T)
(PROG (CHAR COMMAND DIR OBJ ACCESS USER PWD)
PARSE
(FRESHLINE T)
(printout T "FS: ")
(SETQ CHAR (CHCON1 (READC T T)))
(SELCHARQ CHAR
((L l)
(printout T "ogin")
(SETQQ COMMAND L)
(GO LOGIN))
((S s)
(printout T "how")
(SETQQ COMMAND S))
((A a)
(printout T "dd")
(SETQQ COMMAND A))
((R r)
(printout T "emove")
(SETQQ COMMAND R))
((C c)
(printout T "hange")
(SETQQ COMMAND C))
((P p)
(printout T "ush (type OK to return)")
(RETURN (LIST 'P)))
((^Y)
(printout T " (type OK to return)")
(RETURN (LIST 'P)))
```

```

((Q q)
  (printout T "uit")
  (RETURN (LIST 'Q)))
((H h ?)
  (printout T " one of:" T " L - Login," T " S - Show," T " A - Add," T " R -
  Remove," T " C - Change," T " P, ^Y - Push," T "or Q - Quit.")
  (GO PARSE))
((CR LF)
  (GO PARSE))
(PROGN (PRIN1 (CHARACTER (CHARCODE ^G)))
  (GO PARSE))
(SETQ DIR (ACCESS.PARSE.DIRECTORY DIR.DEFAULT USER/PWD))
(OR DIR (RETURN ' (ABORT)))
[SELECTQ COMMAND
  (S
    (* done w/ List parse)
    (RETURN (LIST COMMAND DIR)))
  (PROGN (SETQ OBJ (ACCESS.PARSE.OBJ OBJ.DEFAULT COMMAND))
    (OR OBJ (RETURN ' (ABORT)))
    (SELECTQ COMMAND
      (R
        (* done w/ Remove parse)
        (RETURN (LIST COMMAND DIR OBJ)))
      (PROGN (SETQ ACCESS (ACCESS.PARSE.ACCESS ACCESS.DEFAULT COMMAND))
        (if ACCESS
          then (RETURN (LIST COMMAND DIR OBJ ACCESS))
          else (RETURN ' (ABORT)))
        (* done w/ add parse)
      )
    )
]
LOGIN
[SETQ USER (MKATOM (PROMPTFORWARD " (username)" (CAR USER/PWD)
  NIL NIL NIL NIL (LIST (CHARCODE CR)
  (CHARCODE LF]
  (OR USER (RETURN ' (ABORT)))
  (SETQ PWD (\ENCRYPT.PWD (PROMPTFORWARD " (password)" NIL NIL NIL "**)))
  (OR PWD (RETURN ' (ABORT)))
  (RETURN (CONS COMMAND (CONS USER PWD]))

```

(ACCESS.PARSE.DIRECTORY

```

[LAMBDA (DIR.DEFAULT USER/PWD DON'T.CHECK)
  (* drc: " 1-Apr-86 16:13")
  (DECLARE (GLOBALVARS \CONNECTED.DIRECTORY))
  (LET ((DIR (PROMPTFORWARD " (access to directory)" DIR.DEFAULT "the name of an NS directory")))
    (if (NOT DIR)
      then
        (* user just typed CR)
        NIL
      else
        (* default host to connected host)
        [OR (FILENAMEFIELD DIR 'HOST)
          (SETQ DIR (MKSTRING (PACKFILENAME 'HOST (FILENAMEFIELD \CONNECTED.DIRECTORY 'HOST)
          'DIRECTORY DIR]
        (if DON'T.CHECK
          then DIR
          elseif (ACCESS.DIRECTORYNAME DIR USER/PWD)
          else (printout T " not an NS directory.")
          NIL])

```

(ACCESS.DIRECTORYNAME

```

[LAMBDA (HOST/DIR USER/PWD)
  (* drc: " 1-Apr-86 16:35")
  (if (AND (STRPOS ":" HOST/DIR)
    (DIRECTORYNAMEP HOST/DIR))
    then HOST/DIR
    elseif [LET ((POS (STRPOS ">" HOST/DIR))
      (AND POS (STRPOS ">" HOST/DIR (ADD1 POS)
      then (SELECTQ (ASKUSER DWIMWAIT 'Y (CONCAT " Create subdirectory " HOST/DIR " ? "))
        (Y (ACCESS.MKDIR HOST/DIR USER/PWD))
        NIL])

```

(ACCESS.MKDIR

```

[LAMBDA (HOST/DIR USER/PWD)
  (* drc: " 1-Apr-86 16:54")
  (RESETLST
    (LET* ((HOST (FILENAMEFIELD HOST/DIR 'HOST))
      (DIR (ACCESS.SLASHIFY.DIRNAME (FILENAMEFIELD HOST/DIR 'DIRECTORY)))
      [PARENT (CONCATLIST (DREVERSE (CDR (FMEMB '/' (DREVERSE (UNPACK DIR)
      (CONNECTION (ACCESS.OPEN HOST PARENT USER/PWD))
      (STREAM (CAR CONNECTION))
      (SESSION (CADR CONNECTION))
      (HANDLE (CADDR CONNECTION))
      (COURIER.CALL STREAM 'FILING 'CREATE HANDLE (BQUOTE ((NAME , DIR)
        (IS.DIRECTORY T)
        (FILE.TYPE 1)))
        NIL SESSION 'RETURNERRORS)
      HOST/DIR))])

```

(ACCESS.PARSE.OBJ

```

[LAMBDA (OBJ.DEFAULT COMMAND)
  (* drc: "28-Mar-86 13:24")

```

```
(LET [(OBJ (PROMPTFORWORD (CONCAT " (" (SELECTQ COMMAND
                                (C "for ")
                                ""))
                                "user or group"))
      (OBJ.DEFAULT "an NS user or group name" NIL NIL NIL (LIST (CHARCODE CR)
                                                                (CHARCODE LF))
      (if (NOT OBJ)
          then
              NIL
          else (if (CH.LOOKUP.OBJECT OBJ)
                  else (printout T " not an NS object."
                                NIL]))
      (* user typed CR)
```

(ACCESS.PARSE.ACCESS

```
[LAMBDA (ACCESS.DEFAULT COMMAND)
  (LET ((ACCESS (PROMPTFORWORD (SELECTQ COMMAND
                                (C " (to be) ")
                                (A " (with access) ")
                                (SHOULDNT "UNKNOWN COMMAND"))
                                (AND ACCESS.DEFAULT (ACCESS.NUM.TO.STRING ACCESS.DEFAULT))
                                "A sequence of letters (R=Read, W=Write, A=Add, D=Delete, C=Change access list)"))
        (ACCESS.BYTE NIL))
      (if (NOT ACCESS)
          then
              NIL
          else (SETQ ACCESS.BYTE (ACCESS.STRING.TO.NUM ACCESS))
              (if (AND ACCESS.BYTE (IGREATERP ACCESS.BYTE 0)
                    (ILESSP ACCESS.BYTE 32))
                  then ACCESS.BYTE
                  else (printout T " bad access specification."
                                NIL]))
      (* drc: "28-Mar-86 13:25")
      (* user just typed CR)
```

(ACCESS.DO.COMMAND

```
[LAMBDA (USER/PWD COMMAND HOST/DIR NSNAME ACCESS)
  (* Performs COMMAND (one of S, A, R, or C) * HOST/DIR should be an NS host & dir, NSNAME should be a valid NS
  name (not used for S command), ACCESS should be an integer between 1 and 31
  (not used for S or R commands)%.)
  (RESETLST
   (PROG ((HOST (FILENAMEFIELD HOST/DIR 'HOST))
          (DIR (FILENAMEFIELD HOST/DIR 'DIRECTORY))
          CONNECTION STREAM SESSION HANDLE OLD.LIST)
         (SETQ CONNECTION (ACCESS.OPEN HOST DIR USER/PWD))
         (SETQ STREAM (CAR CONNECTION))
         (SETQ SESSION (CADR CONNECTION))
         (SETQ HANDLE (CADDR CONNECTION))
         (SETQ OLD.LIST (CAADAR (COURIER.CALL STREAM 'FILING 'GET.ATTRIBUTES HANDLE (LIST 19)
                                                                    SESSION)))
         (* list of triples ala (NSNAME GroupOrIndividual Access#))
         (SELECTQ COMMAND
          (S (ACCESS.SHOW HOST/DIR OLD.LIST))
          (R (ACCESS.REMOVE NSNAME HOST/DIR OLD.LIST STREAM HANDLE SESSION))
          (A (ACCESS.ADD NSNAME HOST/DIR ACCESS OLD.LIST STREAM HANDLE SESSION))
          (C (ACCESS.CHANGE NSNAME HOST/DIR ACCESS OLD.LIST STREAM HANDLE SESSION))
          (SHOULDNT))))))
  (* drc: " 1-Apr-86 16:37")
```

(ACCESS.OPEN

```
[LAMBDA (HOST DIR USER/PWD)
  (* returns a list of a courier stream and a courier session on HOST for USER/PWD.
  If DIR is NON-nil, will also return a handle for it. Note that this expects to be called from within a RESETLST.)
  (LET ((CREDENTIALS (NS.MAKE.SIMPLE.CREDENTIALS USER/PWD))
        (STREAM (COURIER.OPEN HOST NIL NIL (CONCAT HOST " Access")))
        SESSION HANDLE)
      (RESETSAVE NIL (LIST 'CLOSEF? STREAM))
      (SETQ SESSION (COURIER.CALL STREAM 'FILING 'LOGON (PARSE.NSNAME HOST)
                                         (CAR CREDENTIALS)
                                         (CDR CREDENTIALS)))
      (if DIR
          then (SETQ HANDLE (COURIER.CALL STREAM 'FILING 'OPEN (LIST (LIST 'PATHNAME (ACCESS.SLASHIFY.DIRNAME
                                                                                   DIR))
                                                                    \NSFILING.NULL.HANDLE NIL SESSION))
                    (RESETSAVE NIL (LIST 'COURIER.CALL STREAM 'FILING 'CLOSE HANDLE SESSION T)))
          (LIST STREAM SESSION HANDLE))
      (* drc: " 1-Apr-86 16:26")
```

(ACCESS.SHOW

```
[LAMBDA (HOST/DIR L)
  (printout T T "Access list for " HOST/DIR ":")
  (for X in L do (printout T T " " (CAR X)
                            .TAB 20 " (" (L-CASE (CADR X))
                            " with "
                            (ACCESS.NUM.TO.STRING (CADDR X)))
  (* edited: "11-Mar-86 11:01")
```


{MEDLEY}<lispusers>ACCESS.;1 (ACCESS.STRING.TO.NUM cont.)

Page 5

(A 2)
(C 4)
(W 8)
(R 16)
(RETURN NIL)

)
(PUTPROPS **ACCESS COPYRIGHT** ("Speech Input Project, Univ. of Edinburgh" 1986))

FUNCTION INDEX

ACCESS	1	ACCESS.DO.COMMAND	3	ACCESS.PARSE	1	ACCESS.REMOVE	4
ACCESS.ADD	4	ACCESS.MKDIR	2	ACCESS.PARSE.ACCESS	3	ACCESS.SHOW	3
ACCESS.CHANGE	4	ACCESS.NUM.TO.STRING	4	ACCESS.PARSE.DIRECTORY	2	ACCESS.SLASHIFY.DIRNAME	4
ACCESS.DIRECTORYNAME	2	ACCESS.OPEN	3	ACCESS.PARSE.OBJ	2	ACCESS.STRING.TO.NUM	4
